

UT 6.1: Operaciones generales en sistemas Linux

ESTRUCTURA DE DIRECTORIOS FHS

Las distribuciones de Linux tienen un árbol jerárquico o estructura de directorios idéntico, similar a la de UNIX.

FHS establece que la estructura de directorios de Linux comienza desde el directorio superior: la raíz (/), en el que se encuentra el resto de archivos y directorios del sistema.

Cualquier persona o institución puede crear una distribución Linux, pero para ser considerada Linux debe cumplir con el estándar FHS (Filesystem Hierarchy Standard).

- **Estáticos:** Modificables solo por el administrador, contienen archivos como binarios y bibliotecas.
- **Dinámicos:** Pueden ser modificados libremente, contienen documentos y configuraciones de los usuarios.
- **Compartidos:** Archivos compartidos entre uno o varios usuarios del sistema Linux.

En Linux, todo en el sistema se representa como archivos, tanto el software como el hardware. Los dispositivos de almacenamiento, como discos ópticos o USB, se montan como subdirectorios en el sistema de archivos, mostrando su contenido cuando están montados y quedando vacíos al desmontarse.

Directorios:

/: El directorio raíz en Linux es el nivel más alto de la jerarquía de archivos, similar a "C:" en Windows.

/bin/: Binarios, los ejecutables del sistema como ls, cp, mkdir, similar a los .exe en Windows.

/boot/: Archivos del cargador de arranque (GRUB, LILO) y el kernel.

/dev/: Representación de hardware como micrófonos, impresoras y USB, incluso si no están montados.

/proc/: Enlaces a procesos en ejecución y datos sobre la actividad interna de Linux.

/etc/: Almacena los principales archivos de configuración del sistema y scripts de inicio.

/home/: Contiene archivos personales y configuraciones de los usuarios, excepto el superusuario (*root*), que tiene su propio directorio.

/lib/: Guarda bibliotecas esenciales, módulos del núcleo y drivers.

/mnt/: Para montajes temporales de unidades, similar a **/media**, pero destinado a usuarios.

/sbin/: Contiene binarios y comandos exclusivos para el superusuario (*root*).

/srv/: Almacena datos de servicios en red, como servidores web, FTP o HTTP.

/tmp/: Directorio para archivos temporales, que se limpia en cada reinicio.

/usr/: Contiene programas no esenciales y archivos compartidos de solo lectura para todos los usuarios.

/root/: Carpeta personal del superusuario (*root*), separada de **/home/**.

/var/: Guarda archivos dinámicos como logs, correos electrónicos y bases de datos.

VARIABLES DE ENTORNO

Las variables de entorno en Bash almacenan datos en memoria y se identifican con un nombre alfanumérico. Se acceden con el prefijo \$ y se pueden listar con env. Algunas importantes son:

- **\$HOME**: Directorio personal del usuario.
- **\$HOSTNAME**: Nombre del equipo.
- **\$PATH**: Directorios donde se buscan los ejecutables. Ruta donde están todos los programas.
- **\$SHELL**: Intérprete de comandos predeterminado.
- **\$TERM**: Tipo de terminal.
- **\$USER**: Nombre del usuario.
- **\$PWD**: Directorio actual.
- **\$PS1**: Prompt principal.
- **\$RANDOM**: Genera un número aleatorio.

El comando alias en Linux permite reemplazar una cadena por otra al ejecutar comandos en la terminal.

Algunas características clave son:

- Los alias solo persisten durante la sesión actual.
- Pueden hacerse permanentes al modificarse el archivo .bash_profile.
- Para ver todos los comandos con alias se pone: alias
- Para crear un nuevo alias, se pone:
 - alias nombre="lista_comando_parámetros"

Modificación de Variables:

Las variables en Bash pueden modificarse o crear nuevas según sea necesario.

- Para crear una variable: **\$HOY=viernes**
- Para ver su valor: **echo \$HOY**
- Las variables locales aparecen con set, pero no con env.
- Para hacerla accesible a subrutinas o procesos hijos: **export HOY**
- Para que sea permanente, debe añadirse a un perfil, como **/etc/profile**.
- Para eliminarla: **unset HOY**

EDITORES DE TEXTO EN LINUX

Los editores de texto principales en Linux son:

nano	<p>Nano es uno de los editores de texto más usados en la terminal de Linux debido a su simplicidad, facilidad de uso y preinstalación en muchas distribuciones.</p> <ul style="list-style-type: none"> • Los comandos en Nano se ejecutan con combinaciones de Ctrl + una letra. • En la parte inferior de la pantalla se muestran los atajos disponibles. • Es ideal para editar archivos de configuración y textos sin necesidad de interfaz gráfica.
vi / vim	<p>Vim es una versión mejorada del antiguo Vi, es flexible y potente en la edición de texto.</p> <ul style="list-style-type: none"> • Aunque tiene una curva de aprendizaje moderada, es muy usado por desarrolladores y usuarios avanzados. • Permite expresiones regulares para buscar y reemplazar texto. • Facilita la navegación rápida entre líneas, párrafos y palabras. • Soporta resaltado de sintaxis y muchas opciones avanzadas de edición.
emacs	<p>Editor de comando más antiguo y versátil para sistemas basados en Linux y UNIX, y bastante complejo:</p> <ul style="list-style-type: none"> • Proporciona un entorno integrado para diferentes tipos de tareas. • Permite cambiar a modo trabajo similar a Vi / Vim.
gedit	<p>Un clásico y popular editor de textos sencillo pero potente para entornos gráficos basados en Gnome.</p> <ul style="list-style-type: none"> • Existen ports para otros SO como Windows o Mac. • Incluye herramientas para la edición de código fuente y textos estructurados, como lenguajes de marcado.

METACARACTERES

En Bash, los metacaracteres están reservados para **scripts y comandos**:

?	Representa un único carácter.
*	Representa ningún o cualquier número de caracteres.
[abc]	Coincide con cualquiera de los caracteres dentro del conjunto (a, b o c).
[a-z]	Coincide con cualquier carácter dentro del rango (de 'a' a 'z').
{a,b,c}	Coincide con cualquiera de los patrones separados por comas.
[!abc]	Excluye caracteres específicos (no a, b ni c).
 	Alternativa entre dos expresiones.
\	Escapa un metacaracter para mostrarlo sin interpretarlo.

Las expresiones regulares se utilizan para describir patrones de texto.

. *	Para representar cero o más caracteres.
. (punto)	Para representar un solo carácter
[aeiou]	Para representar algunos caracteres
[A-Z][0-9]	Para presentar un rango de caracteres
^	Para representar el inicio de una línea de texto
\$	Para representar el fin de línea de texto o de la palabra

OPERACIONES GENERALES

Repaso de comandos básicos de utilización en la consola/terminal de Linux:

pwd	Muestra el directorio en el que nos encontramos	pwd
man	Muestra ayuda sobre un comando	man ls
env	Muestra la lista de variables de entorno	env
cd	Cambia de directorio	cd mp3/wim_mertens
touch	Crea un fichero vacío	touch listado.txt
mkdir	Crea uno o varios directorios	mkdir cartas facturas
cp	Copia archivos o directorios	cp *.txt directorio/
rm	Borra archivos o directorios	rm -R file.tmp
mv	Mueve o renombra archivos o directorios	mv fil.txt texto.txt
rmdir	Borra directorios	rmdir Viejo
ls	Lista ficheros y directorios	ls -l
cat	Visualiza un fichero	cat /var/log/dmesg
less	Visualiza un fichero pantalla a pantalla y permite retroceder	less /var/log/dmesg
head	Muestra las primeras líneas de un fichero	head -n5 /var/log/dmesg
tail	Muestra las últimas líneas de un fichero	tail /var/log/dmesg
date	Muestra la fecha/hora actuales en un formato concreto	date
nano	Editor de texto simple	nano listado.txt
vi	Editor de texto potente	vi listado.txt

Los comandos en negrita son para operaciones generales de manejo de directorios y ficheros.

Cambio de Usuario

En sistemas Linux, para cambiar de usuario o usar privilegios de administrador (root), se utilizan los comandos **su** y **sudo**.

- **su** (Switch User) permite cambiar de usuario, mientras que **sudo** (Switch User Do) permite ejecutar comandos con privilegios de administrador de manera segura.

En Ubuntu, la cuenta **root** está desactivada por defecto por razones de seguridad, y se usa un grupo de usuarios llamado **sudoers** (administradores), que pueden ejecutar comandos con permisos de root usando **sudo**.

- **su nombreusuario**: Cambia al usuario especificado.
- **sudo comando**: Ejecuta un comando con privilegios de root.

Para programas gráficos que requieran permisos de administrador, se usa el comando **gksudo**.

Ficheros

ls: Muestra la lista de archivos y directorios en el directorio actual.

ls [directorio | fichero]: Muestra los archivos en el directorio especificado o el contenido de un fichero.

Parámetros:

- **-l**: Muestra información detallada de los archivos (como permisos, propietario, tamaño, etc.).
- **-a**: Muestra todos los archivos, incluidos los ocultos (que empiezan con un punto).
- **-t**: Ordena los archivos por fecha de modificación.
- **-r**: Ordena los archivos en orden inverso.

Nota: **ls -l** puede abreviarse como **ll**.

- **cat**: Muestra el contenido completo de un fichero de texto en la terminal.
- **head**: Muestra las primeras líneas de un fichero.
- **tail**: Muestra las últimas líneas de un fichero.

Date

El comando **date** en Linux se utiliza para mostrar la fecha y la hora actual. Además, permite formatear la salida con una secuencia de caracteres de control precedidos por el signo **+**. Los caracteres de formato comienzan con **%** y se sustituyen por el valor correspondiente en la fecha y hora.

- **%d**: Día del mes (con dos dígitos).
- **%H**: Hora en formato de 24 horas (00-23).
- **%m**: Mes (01-12).
- **%M**: Minuto (00-59).
- **%s**: Segundo (00-59).
- **%Y**: Año completo (por ejemplo, 2025).
- **%u**: Día de la semana (1-7, donde 1 es lunes y 7 es domingo).

Enlaces

En Linux, un enlace es una referencia a un archivo o directorio, similar a un acceso directo o puntero. Existen dos tipos de enlaces:

1. **Enlaces simbólicos (blandos)**: Actúan como accesos directos a archivos o directorios, incluso en otras particiones. Se crean con `ln -s` y permiten redirigir al archivo o directorio original. Si el archivo original se borra, el enlace simbólico se rompe.
2. **Enlaces duros (físicos)**: Asignan un nuevo nombre en un directorio a un archivo o inodo existente, compartiendo el mismo inodo. Los cambios realizados en uno afectan a todos los enlaces duros. Si se borra el archivo original, los enlaces duros siguen funcionando. Solo pueden crearse en la misma partición y se crean con `ln`.

Los sistemas de archivos usados por Linux identifican los archivos en el disco mediante un número llamado inodo. Para conocer el número de inodo de un archivo, se utiliza `ls -li`.

Ficheros Ocultos

En Linux, ocultar archivos y directorios es sencillo. La forma más común es agregar un punto (.) al inicio del nombre del archivo o directorio. Otra opción es crear un archivo llamado `.hidden`, en el que se colocan los nombres de los archivos y directorios que se desean ocultar. Para ver archivos ocultos, se usa el comando `ls -a`.